REAL TIME COMMUNICATION SYSTEM FOR SPECIALLY ABLED USING AI

PROJECT REPORT

1. **INTRODUCTION**
    1. Project Overview
    2. Purpose
2. **LITERATURE SURVEY**
    1. Existing problem
    2. References
    3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    1. Empathy Map Canvas
    2. Ideation & Brainstorming
    3. Proposed Solution
    4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    1. Functional requirement
    2. Non-Functional requirements
5. **PROJECT DESIGN**
    1. Data Flow Diagrams
    2. Solution & Technical Architecture
    3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    1. Sprint Planning & Estimation
    2. Sprint Delivery Schedule
    3. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    1. Feature 1
    2. Feature 2

Database Schema (if Applicable)

8. **TESTING**
    1. Test Cases
    2. User Acceptance Testing
9. **RESULTS**
    1. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code
GitHub & Project Demo Link

1. INTRODUCTION

1. Project Overview

Communication is a social process of exchanging information from one entity to another in verbal and non-verbal form. It defines our existence and it is an important instrument that connects people together. It comes naturally as a raw skill embedded in most people at birth and we acquired the ways of communication through cognitive learning. Communication is the basis, which drives the process of development in all the fields and it is the very core of our civilization. The ability to communicate allows us to express emotion, feelings, convey our thoughts and ideas as well as to relate our experiences. It plays an important role in the dissemination of information and sharing of knowledge especially in the academic arena. Research has found that humans started to learn how to communicate with each other since they are born not only through spoken and written languages but also body gesture, posture, facial expression and eye contacts. Communication skill might come as a natural ability in the majority of people.

2. Purpose

Many assistive tools or formally termed as Alternative and Augmentative Communication has been developed and employed to assist people with impaired communication skills. The term encompasses the whole combination of methods used for communication such as text to speech system, pointing gestures, facial expression and body language. Although these AACs have been widely used to assist the disabled, it is not potentially effective because most AACs are text to speech and touch screen based applications, which are unsuitable for those with severe physical abilities.

2. LITERATURE SURVEY

1. Existing problem

There are some people afflicted with some form of physical defects which affect their ability to communicate. One of the more severe disabilities is known as “cerebral palsy”, a congenital disorder at birth which causes abnormality in their motor system. It affects their muscle movement and coordination, learning and speech abilities. Their malfunctioned motor system causes an uncontrollable and involuntary movement. They are unable to control their oral- facial muscles, thus affecting their ability to perform facial expression appropriately.

It’s been over a decade since facial recognition technology has been a significant topic in the news. It made headlines in 2005 when it was used to identify the 9/11 terrorists. Ten years later, it seems that this high-tech innovation is being used less for catching criminals and more for making people feel secure.

2. References

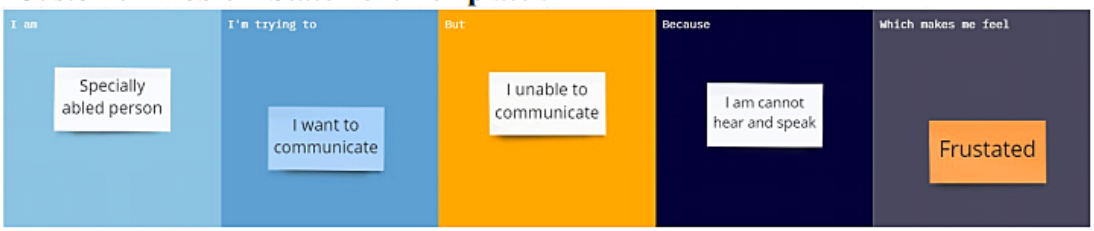
1. Prof. P.G. Ahire, K.B. Tilekary,T.A. Jawake, P.B. Warale, “Two Way Communicator between Deaf and Dumb People and Normal People", 978-1-4799-6892-3/15 31.00 c 2015 IEEE. 3
2. Shreyashi Narayan Sawant, "Sign Language recognition System to aid Deaf- dumb People Using PCA", IJCSET ISSN : 2229-3345 Vol. 5 No. 05 May 2014.
3. Amitkumar Shinde, Ramesh Kagalkar, ”Sign Language to Text and Vice Versa Recognition using Computer Vision in Marathi”, International Journal of Computer Applications (0975 – 8887) National Conference on Advances in Computing (NCAC 2015)
4. M. Ebrahim Al-Ahdal & Nooritawati Md Tahir,’’ Review in Sign Language Recognition Systems’’ Symposium on Computer & Informatics(ISCI),pp:52-57, IEEE ,2012

3. Problem Statement Definition

In our society, we have people with disabilities. The technology is developing day by day but no significant developments are undertaken for the betterment of these people.

Communication between deaf-mute and a normal person has always been a challenging task. It is very difficult for mute people to convey their message to normal people. Since normal people are not trained in hand sign language. In emergency times conveying their message is very difficult. The human hand has remained a popular choice to convey information in situations where other forms like speech cannot be used. Voice Conversion System with Hand Gesture Recognition and translation will be very useful to have a proper conversation between a normal person and an impaired person in any language. The project aims to develop a system that converts the sign language into a human hearing voice in the desired language to convey a message to normal people, as well as convert speech into understandable sign language for the deaf and dumb. We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model. This app enables deaf and dumb people to convey their information using signs which get converted to human understandable language and speech is given as output.

Customer Problem Statement Template :

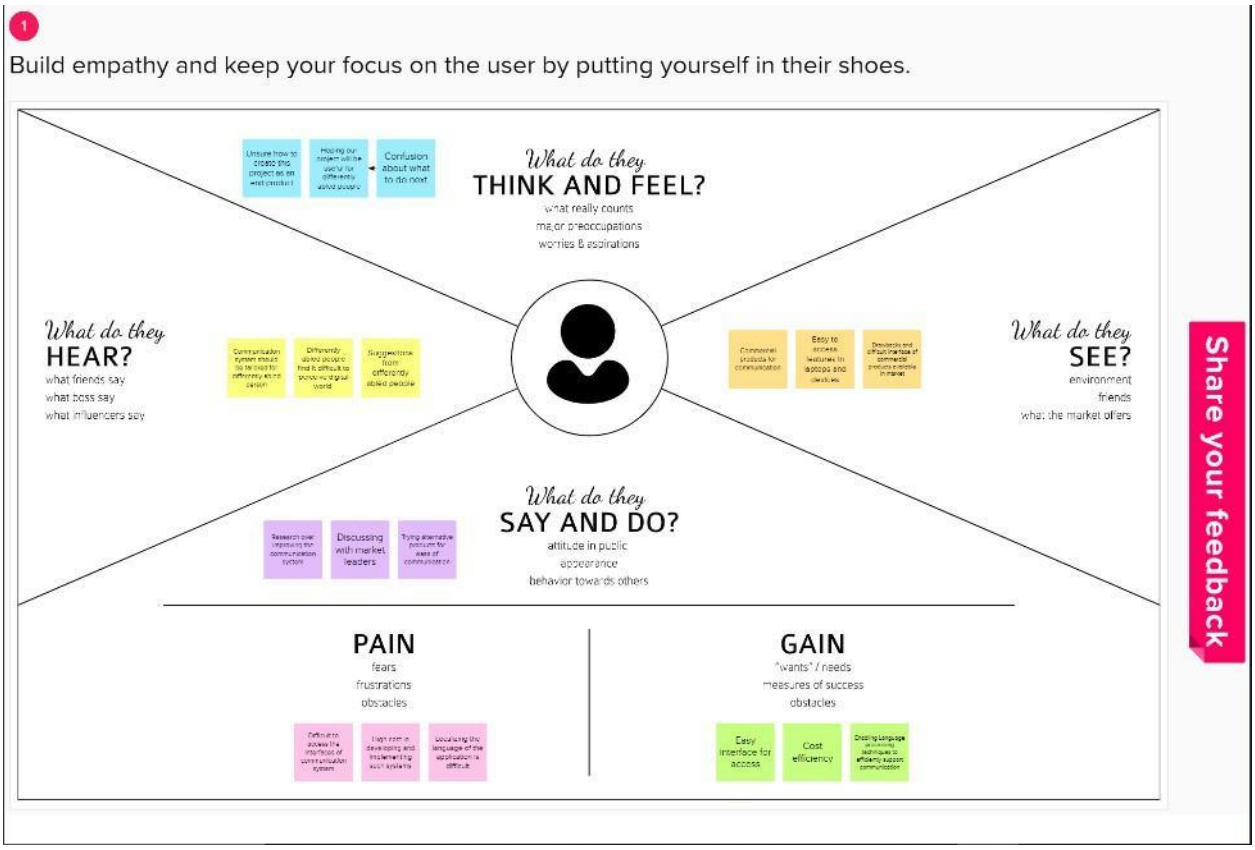


Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Specially abled	hear sound	I unable to hear	I am physically disabled to hear	frustrated

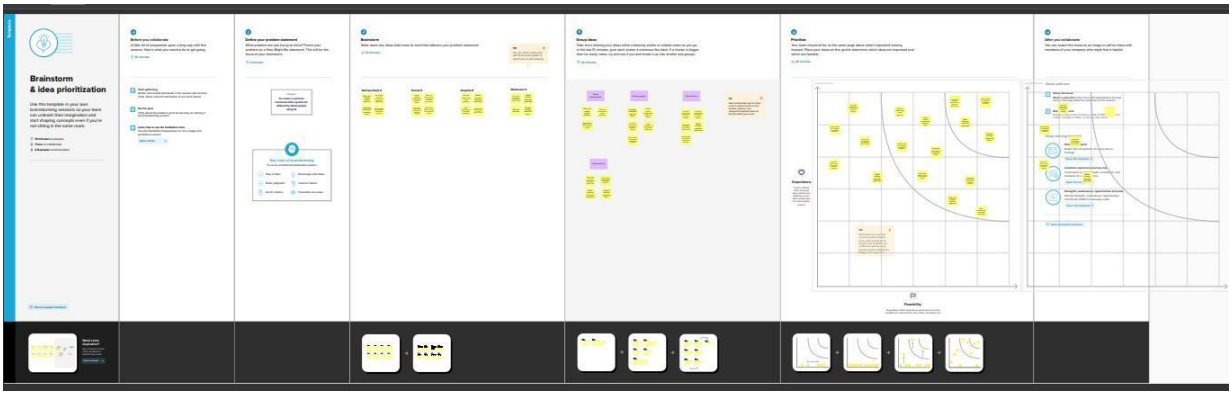
3. IDEATION & PROPOSED SOLUTION

1. Empathy Map Canvas

Empathy map was done by our team using the ideas of how, why and what users think or say about their feelings over this project.



2. Ideation & Brainstorming



3. **Proposed Solution**

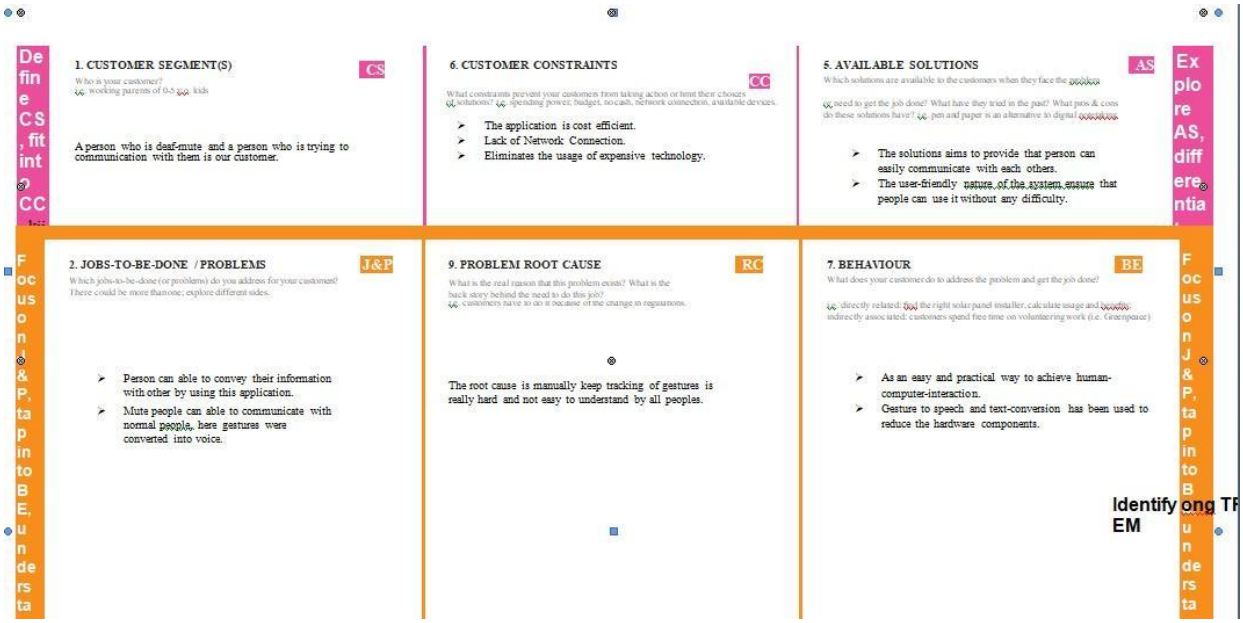
- We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model.

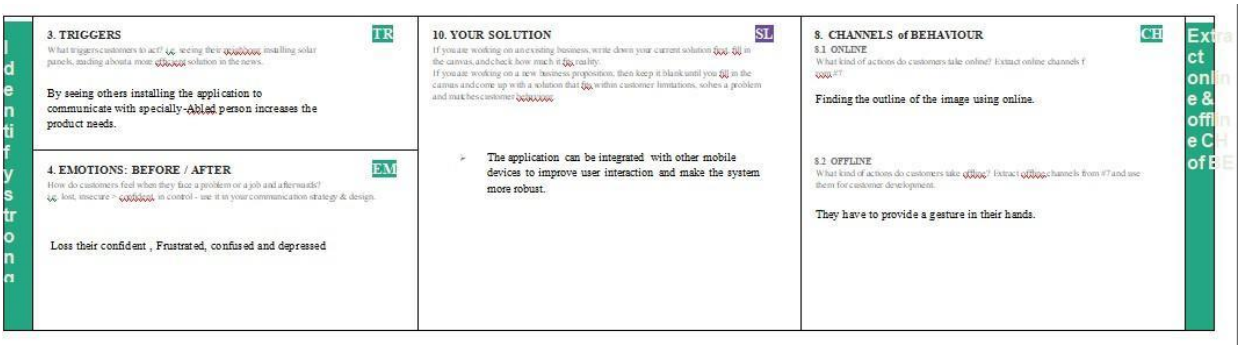
- This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is given as output.
- The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between deaf-mute and a normal person has always been a challenging task.

S.No.	Parameter	Description
●	Problem Statement (Problem to be solved)	● To develop a system that converts the sign language into a human hearing voice in the desired language to convey a message to normal people, as well as convert speech into understandable sign language for the deaf and dumb.
●	Idea / Solution description	● The technology is developing day by day but no significant developments are undertaken for the betterment of these people. Communication between deaf-mute and a normal person has always been a challenging task. ● We are making use of a convolution neural network to create a model that is trained on different hand gestures. An app is built which uses this model. ● This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is given as output.
●	Novelty / Uniqueness	● Cost effective. ● Reduces delay between gesture recognition. ● Provides a technique for dumb people using text to voice conversion.
●	Social Impact / Customer Satisfaction	● This project is helpful in providing a great solution for many deaf and dumb persons . ● This helps the specially abled people to maintain a life balance with others.
●	Business Model (Revenue Model)	● Helps the companies in project scheduling. ● Usable by many differently abled which increases revenue.

•	Scalability of the Solution	• Made available for many specially abled people. • Dataset can be updated according to future needs.

4. **Problem Solution fit**





4. REQUIREMENT ANALYSIS

4.1 Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Authentication	Authentication through Facial recognition Authentication through Password authentication protocol
FR-4	External interfaces	Robots and other tools provide home-based care and other assistance, allowing people with disabilities to live independently
FR-5	Transaction Processing	More application can use to translate the sign language like D talk in the system
FR-6	Reporting	There is a growing feeling that we need to do more, to help make the lives of people with disabilities easier
FR-7	Business rules	Human augmentation and Practical accuracy are responsible for AI business rules

2. Non-Functional requirements

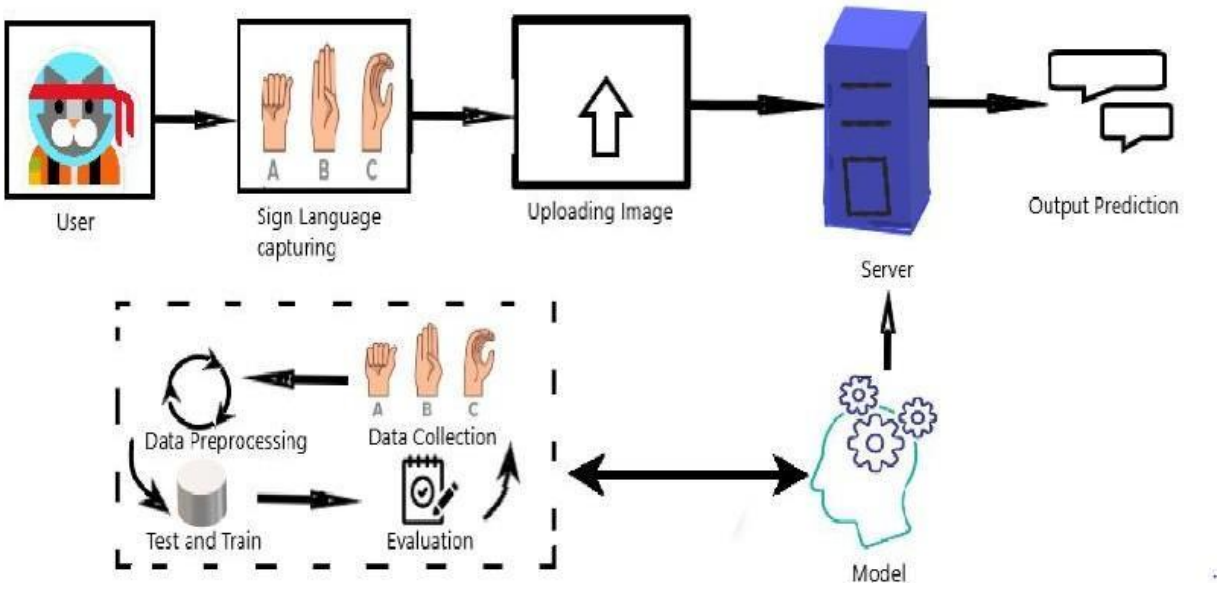
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	provide personalised learning experiences tailored to the specific needs of students with disabilities
NFR-2	**Security**	Set the inclusion and exclusion criteria , Report the results in the survey
NFR-3	**Reliability**	It setting the pace of the future and helping people in need
NFR-4	**Performance**	enables people with disabilities to step into a world

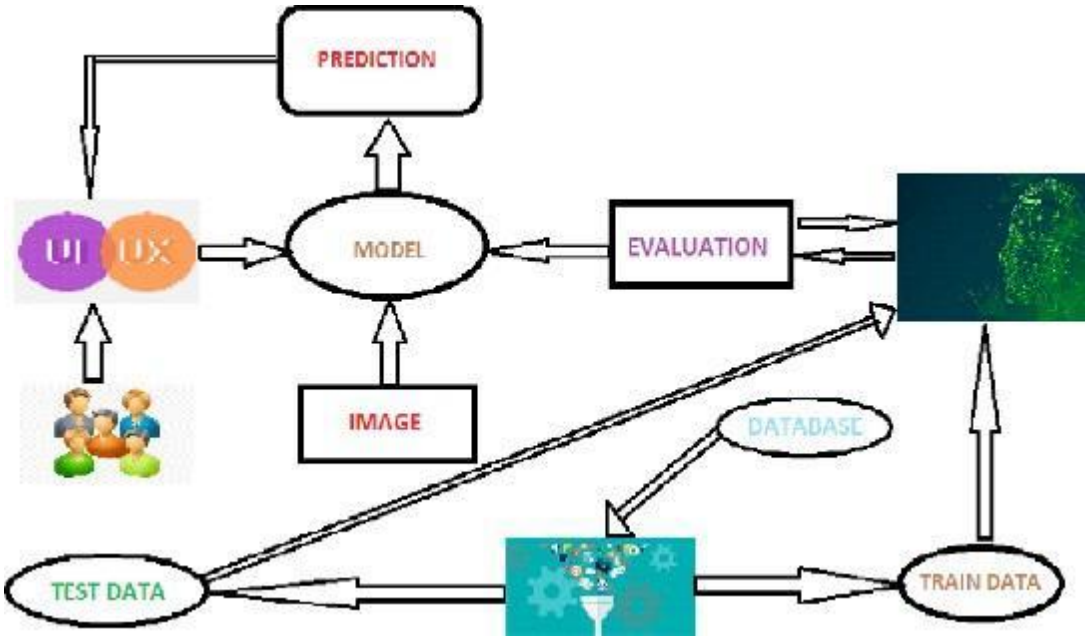
		where their difficulties are understood and taken into account
NFR-5	**Availability**	Technology solutions that mimic humans and use logic from playing chess to solving equations and Machine learning is one of the technologies
NFR-6	**Scalability**	The improvement in the specially abled persons interaction with the environments

5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



3. User Stories

- User can register for the application by entering their email, password, and confirming their password.
- User can register for the application through Google.
- User can receive confirmation content through email.
- User can log into the application by entering email & password.
- User can register by giving in their email, password and confirmation of password.
- User can log into the application by entering email & password.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customers (Mobile user)	Registration	USN-1	User can register for the application by entering their email,password, and confirming their password.	Users will be able to access their account	High	Sprint-1
		USN-2	User can register for the application through Google	receiving confirmation email	Low	Sprint-2
		USN-3	User can receive confirmation content through email	Registration gets completed here and user can access their account.	High	Sprint-1
	Login	USN-4	User can log into the application by entering email & password		High	Sprint-1

Customers (Web user)	Registration	USN-1	User can register by giving in their email, password and confirmation of password.	User can access their account	High	Sprint-1
	Login	USN-2	User can log into the application by entering email & password		High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Pre-Requisites	M-01	To complete this project we should have known the following software concepts and packages such as Keras, Tensorflow, Python, Anaconda, OpenCV, Flask,etc…	Yes
Project Structure	M-02	To build a Project Structure which needs to be followed for building Conversation Engine	Yes
Collection of Data	M-03	To collect data for the image Preprocessing, Collection of dataset aredivided into Train set and Test set, The two files are compiled separately and executed for the Training and Testing process.	Yes
Image Preprocessing	M-04	Importing the Image Data Generator libraries, Applying Image Data Generator Functionality to train set and test set	Yes

Model Building	M-05	Importing the model building libraries, Initializing the model, Adding Convolution layers, Adding the Pooling layers, Adding the Flatten layers, Adding Dense layers, Compiling the model , Fit and Save the model.	Yes
Test the model	M-06	Import the packages and save the model and Load the test image, pre- process it and predict it.	Yes

Customer Problem Statement Template :

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Specially abled	hear sound	I unable to hear	I am physically disabled to hear	frustrated

Application layer	M-07	Build the flask application and the HTML pages.
Train CNN model	M-08	Register for IBM Cloud and train Image Classification Model
Ideation Phase	M-09	Preparation of Literature Survey, Information Gathering, empathy map and ideation
Project Design Phase-I	M-10	Preparation of Proposed solution, Problem-Solution fit and Solution Architecture
Project Design Phase-II	M-11	Preparation of Customer journey map, Functional requirements, Data flow diagram and Technology stack Architecture
Project Planning Phase	M-12	Prepare Milestone list , Activity list and Sprint Delivery Plan
Project Development Phase	M-13	Project Development delivery of Sprint 1, Sprint 2, Sprint 3,Sprint 4

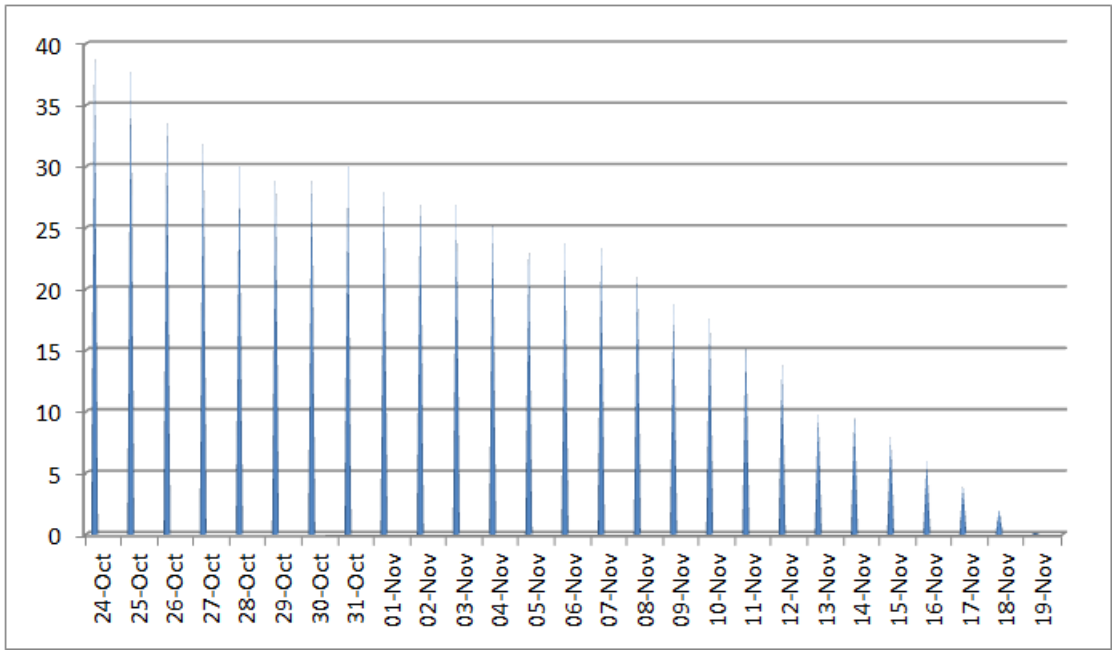
6.2 Sprint delivery schedule:

Project Tracker, Velocity &Burn down Chart:|

Sprint	Total StoryPoints	Duration	Sprint Start Date	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	6Days	24Oct 2022	28Oct 2022	20	28Oct 2022
Sprint-2	15	6 Days	31Oct 2022	05Nov 2022	20	05Nov 2022
Sprint-3	20	7 Days	07Nov 2022	13Nov 2022	20	13Nov 2022
Sprint-4	20	6Days	14Nov 2022	19Nov 2022	20	19Nov 2022

3. Reports from JIRA

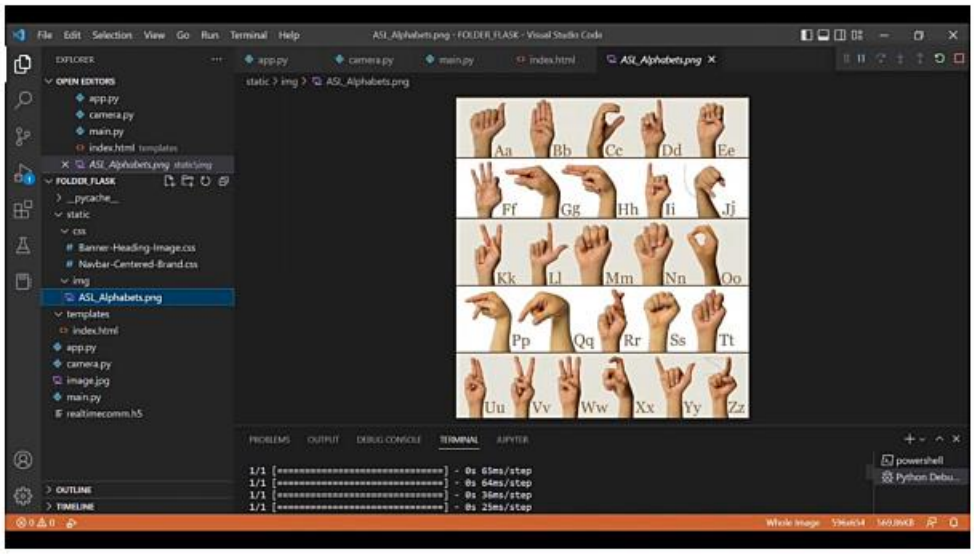
Burn-downChart:



7. CODING & SOLUTIONING (Explain the features added in the project along with code)

1. **Feature 1**

- Real time video input using opencv
- With OpenCV, we are capturing a video from the camera. It lets us create a video capture object which is helpful to capture videos through webcam and then we may perform desired operations on that video.



2. **Feature 2**

- Hand Region Segmentation & Hand Detection and Tracking:
- The captured images are scanned for hand gestures. This is a part of preprocessing before the image is fed to the model to obtain the prediction. The segments containing gestures are made more pronounced. This increases the chances of prediction by many folds.
- The model accumulates the recognized gesture to words. The recognized words are converted into the corresponding speech. The text to speech result is a simple work around but is an invaluable feature as it gives a feel of an actual verbal conversation.

8. TESTING

8.1 Test Cases

	Date	3 Nov 22	
	Team ID	PNT2022TMID06858	
	Project Name	Project -REAL TIME COMMUNICATION SYSTEM POWERED BY AI FOR SPECIALLY ABLED	
	Maximum Marks	4 marks	
Test case ID	**Feature Type**	**Component**	**Test Scenario**
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to access the website on their Devices
LoginPage_TC_OO2	UI	Home Page	Verify the UI elements in Home Page
LoginPage_TC_OO3	Functional	Home page	Verify the action from specially abled person and found the solution for their respective actions
LoginPage_TC_OO4	UI	Login page	Verify the UI elements in Home Page
LoginPage_TC_OO5	Functional	Login page	Verify user is able to access the application

	Test Scenarios
1	Verify user is able to see login page
2	Verify user is able to loginto application or not?
3	Verify user is able to navigate to create your account page?
4	Verify user is able to recovery password
5	Veriify login page elements
	Search
1	Verify user is able to search by entering keywords in search box
2	Verify user is able to see suggestions based on keyword entered in search box
3	Verify user is able to see related auto suggestions displaying based on keyword entered in search box
4	Verify user is able to see no matches found message when no results are matching with entered keyword
5	Verify user is able to see seach detailed page when nothing entered in textbox

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Real-Time Communication System Powered by AI for Specially Abled project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	0	1	0	6
Duplicate	1	0	0	0	1

External	0	2	0	0	2
Fixed	7	0	0	0	7
Not Reproduced	0	1	0	0	1
Skipped	0	0	1	0	1
Won't Fix	0	4	0	0	4
Totals	13	7	2	0	44

3. Test Case Analysis

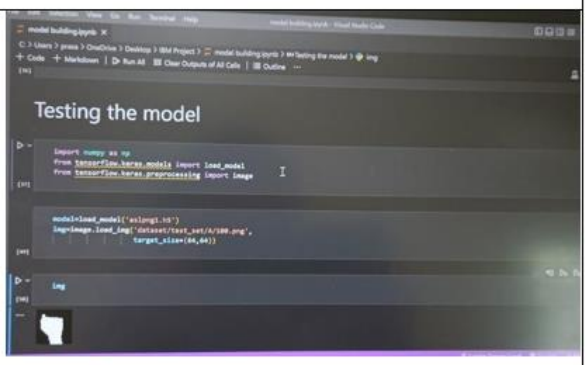
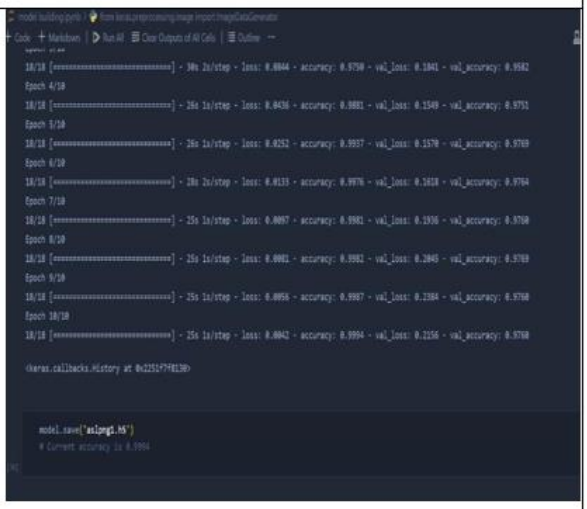
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	9	0	4	5
Client Application	9	0	4	5
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	5	0	0	5
Final Report Output	9	0	4	5
Version Control	2	0	0	2

We had manually done testing of data from the datasets that are present under the folders from A to I. The accuracy and speed of model are found by performing model tests. Here we had defined the set of test data, preconditions, expected results and post conditions, developed for a particular test scenario .

9. RESULTS

Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Model Testing	-	
2.	Accuracy	Training Accuracy - Validation Accuracy -	
.			

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- This app enables deaf and dumb people to convey their information using signs which get converted to human understandable language and speech is given as output.
- This model builds a communication system that enables communications between deaf-dumb person and a normal person.

DISADVANTAGES

- A disadvantage of this model is that it can only predict the letters as of now .The future scope of this model will be to predict and display words as well as sentences through gestures as this model is trained based on predicting only the letter for now.

11 CONCLUSION

The proposed communication system between Deaf and Dumb people and ordinary people are aiming for it when bridging the communication gap between two societies. Several works were done earlier in this area, but this paper adds in complete two - sided communication in an efficient manner because the system is implemented as an easily available application. So, it really serves its needs in all aspects. The above strategies prove to be efficient In terms of time and accuracy.

12 FUTURE SCOPE

The future scope of this model is be to predict and display words as well as sentences through gestures provided by differently abled people. This may enable the communication much easier than how this model could do.

13. APPENDIX

A

P

P

.

```
p
y
from flask import Flask, Response,
render_template from camera import
Video

app =
Flask(
name_)
@app.rout
e('/')
def index():
    return render_template('index.html')

def
gen(
cam
era):
whil
e
True
:
        frame =
        camera.get_fra
        me() yield(b'--
        frame\r\n'
            b'Content-Type:
            image/jpeg\r\n\r\n' + frame +
            b'\r\n\r\n')
```

```
@app.route('/
video_feed')
def
video_feed():
    video = Video()
    return Response(gen(video), mimetype='multipart/x-mixed-replace; boundary =
    frame')


if_name      == '
main_':
app.run()
```

Main.py

```
import cv2
video =


cv2.VideoCaptur


e(0) while True:


    ret, frame =
    video.read()
    cv2.imshow("Fr
    ame", frame) k =
    cv2.waitKey(1)
```

```
        if k == ord('q'):
            break

    video.release
    ()
    cv2.destroyA
    llWindows()
```

Camera.py

```
import cv2
import numpy as np
from keras.models import load_model
from keras.utils import load_img, img_to_array

class Video(object):
def_init_(self):
        self.video =
        cv2.VideoCapture(0)
        self.roi_start = (50,
        150)
        self.roi_end = (250, 350)
        self.model = load_model('aslpng1.h5') # Execute Local Trained Model
        # self.model = load_model('IBM_Communication_Model.h5') # Execute IBM Trained
Model
        self.index=['A','B','C','D','E','F','G','H','I']
        s
elf.y =
None def
del
(self):
        self.vi
deo.release()
def
get_frame(self
):
        ret,frame = self.video.read()
```

```
                frame =
                cv2.resize(frame, (640,
                480)) copy =
                frame.copy()
                copy = copy[150:150+200,50:50+200]
                # Prediction Start
                cv2.imwrite('image.j
                pg',copy)
                copy_img = load_img('image.jpg',
                target_size=(64,64)) x =
                img_to_array(copy_img)
                x = np.expand_dims(x, axis=0)
                pred =
                np.argmax(self.model.predict(x),
                axis=1) self.y = pred[0]
                cv2.putText(frame,'The Predicted Alphabet is:
'+str(self.index[self.y]),(100,50),cv2.FONT_HERSHEY_SIMPLEX,1,(0,0,0),3)
                ret,jpg =
                cv2.imencode('.jpg',
                frame) return
                jpg.tobytes()
```

index.html

```
<!DOCTYP
E html>
        <html lang="en">


        <head>
          <meta charset="utf-8">
          <meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no">
          <title>Sign Language Detection</title>
          <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css">
          <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.12.0/css/all.css">
          <link rel="stylesheet" href="assets/css/Banner-Heading-Image.css">
          <link rel="stylesheet" href="assets/css/Navbar-Centered-Brand.css">
          <link rel="stylesheet" href="assets/css/styles.css">
        </head>
```

```
<body style="background: rgb(247, 246, 244);">
    <nav class="navbar navbar-light navbar-expand-md py-3" style="background: #212529;">
        <div class="container">
            <div></div><a class="navbar-brand d-flex align-items-center" href="#"><span
                    class="bs-icon-sm bs-icon-rounded bs-icon-primary d-flex justify-content-center align-items-center me-2 bs-icon"><i
                        class="fas fa-flask"></i></span><span style="color: rgb(255,255,255);">Real-Time Communication
                    System Powered By AI&amp;nbsp;For Specially Abled</span></a>
            <div></div>
        </div>
    </nav>
    <section>
        <div class="d-flex flex-column justify-content-center align-items-center">
            <div class="d-flex flex-column justify-content-center align-items-center" id="div-video-feed"
                style="width: 640px;height: 480px;margin: 10px;min-height: 480px;min-width: 640px;border-radius: 10px;border: 4px dashed rgb(0, 0, 0) ;">
                <img src="{{ url_for('video_feed') }}" style="width: 100%;height: 100%;color: rgb(0, 0, 0);text-align: center;font-size: 20px;"
                    alt="Camera Access Not Provided!">
            </div>
        </div>
        <div class="d-flex flex-column justify-content-center align-items-center" style="margin-bottom: 10px;"><button
                class="btn btn-info" type="button" data-bs-target="#modal-1" data-bs-toggle="modal">Quick Reference
                -<strong> ASL Alphabets</strong></button></div>
    </section>
    <section>
        <div class="container">
            <div class="accordion text-white" role="tablist" id="accordion-1">
                <div class="accordion-item" style="background: rgb(33,37,41);">
                    <h2 class="accordion-header" role="tab"><button class="accordion-button" data-bs-toggle="collapse"
                            data-bs-target="#accordion-1 .item-1" aria-expanded="true"
                            aria-controls="accordion-1 .item-1"
                            style="background: rgb(39,43,48);color: rgb(255,255,255);">About The Project</button></h2>
                    <div class="accordion-collapse collapse show item-1" role="tabpanel" data-bs-parent="#accordion-1">
                        <div class="accordion-body">
                            <p class="mb-0">Artificial Intelligence has made it possible to handle our daily activities
                                in new and simpler ways. With the ability to automate tasks that normally require human
                                intelligence, such as speech and voice recognition, visual perception, predictive text
```

```
                        functionality, decision-making, and a variety of other tasks, AI can assist people with
                        disabilities by significantly improving their ability to get around and participate in
                        daily activities.<br><br>Currently, Sign Recognition is available <strong>only for
                            alphabets A-I</strong> and not for J-Z, since J-Z alphabets also require Gesture
                        Recognition for them to be able to be predicted correctly to a certain degree of
                        accuracy.</p>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </section>
          <div class="modal fade" role="dialog" tabindex="-1" id="modal-1">
            <div class="modal-dialog" role="document">
              <div class="modal-content">
                <div class="modal-header">
                  <h4 class="modal-title">American Sign Language - Alphabets</h4><button type="button"
                    class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body"><img src="{{ url_for('static', filename='img/ASL_Alphabets.png') }}"
width="100%"></div>
                <div class="modal-footer"><button class="btn btn-secondary" type="button"
                    data-bs-dismiss="modal">Close</button></div>
              </div>
            </div>
          </div>
          <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>

        </body>

        </html>
```

Css

```
.bs-
icon {

        --bs-icon-size: .75rem;
        display: flex;
        flex-shrink: 0;
```

```
  justify-content: center;
  align-items: center;
  font-size: var(--bs-icon-size);
  width: calc(var(--bs-icon-size) * 2);
  height: calc(var(--bs-icon-size) * 2);
  color: var(--bs-primary);
}

.bs-icon-xs {
  --bs-icon-size: 1rem;
  width: calc(var(--bs-icon-size) * 1.5);
  height: calc(var(--bs-icon-size) * 1.5);
}

.bs-icon-sm {
  --bs-icon-size: 1rem;
}

.bs-icon-md {
  --bs-icon-size: 1.5rem;
}

.bs-icon-lg {
  --bs-icon-size: 2rem;
}

.bs-icon-xl {
  --bs-icon-size: 2.5rem;
}

.bs-icon.bs-icon-primary {
  color: var(--bs-white);
  background: var(--bs-primary);
}

.bs-icon.bs-icon-primary-light {
  color: var(--bs-primary);
  background: rgba(var(--bs-primary-rgb), .2);
```

```
}


.bs-icon.bs-icon-semi-white {
  color: var(--bs-primary);
  background: rgba(255, 255, 255, .5);
}


.bs-icon.bs-icon-rounded {
  border-radius: .5rem;
}


.bs-icon.bs-icon-circle {
  border-radius: 50%;
}
```

Image preprocessing

```
from keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale = 1./255, shear_range=0.2,
zoom_range=0.2, horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale = 1./255)

x_train = train_datagen.flow_from_directory('Dataset/training_set',
target_size=(64,64), batch_size = 300, class_mode = 'categorical', color_mode
= 'grayscale')

x_test = test_datagen.flow_from_directory('Dataset/test_set',
target_size=(64,64), batch_size=300, class_mode='categorical', color_mode =
'grayscale')
```

Model Building

```
from keras.models import Sequential

from keras.layers import Dense

from keras.layers import Convolution2D

from keras.layers import MaxPooling2D

from keras.layers import Dropout
```

```
from keras.layers import Flatten

model = Sequential()

model.add(Convolution2D(32,(3,3), input_shape=(64,64,1), activation = 'relu'))

model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Flatten())

model.add(Dense(units=512, activation = 'relu'))

model.add(Dense(units=9, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])

model.fit_generator(x_train, steps_per_epoch=24, epochs=10,
validation_data=x_test, validation_steps=40)
```

Model Testing

In [3]:

```
from keras.models import load_model

import numpy as numpy

import cv2

model=load_model('aslpng1.h5')
```

In [5]:

```
from skimage.transform import resize

def detect(frame):

  img = resize(frame,(64,64,1))

  img = np.expand_dims(img,axis=0)

  if(np.max(img)>1):

    img = img/255.0

  prediction = model.predict(img)
```

```
  print(prediction)

  prediction = np.argmax(prediction,axis=1)

  print(prediction)
```

In [26]:

```
frame=cv2.imread('/content/Dataset/test_set/G/1.png')

data = detect(frame)
```

DATASET DOWNLOAD.ipynb

```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
      "url": "https://us-south.ml.cloud.ibm.com",
      "apikey": "mNVF7E95G-awR213njShj1GiUfN-1SpPq-ko8Wx7na1-"
}

client = APIClient(wml_credentials)

def guid_from_space_name(client, space_name):

    space = client.spaces.get_details()

    return (next(item for item in space['resources'] if item['entity']["name"]
== space_name)['metadata']['id'])

space_uid = guid_from_space_name(client, 'communication_model_deployment')

print("Space UID : ", space_uid)

client.set.default_space(space_uid)

client.repository.download("cefca265-2301-4620-897a-
9c80d6ff7f1a","IBM_Model_Download.tar.gz")
```

Model Building.ipynb

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator

# Training Datagen
train_datagen =
ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_
flip=False)
```

```
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)

import os, types

import pandas as pd

from botocore.client import Config

import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It
includes your credentials.
# You might want to remove those credentials before you share the notebook.
client_9e2ebbcc4db04f3fb5d87e8fa4800e36 = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='GN91c7sTtlR1DYfMzZqIUwGSIkATCCriPItR-s81P82-',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

streaming_body_1 =
client_9e2ebbcc4db04f3fb5d87e8fa4800e36.get_object(Bucket='communicationmodelt
raining-donotdelete-pr-xpzs67frbbb7s3',
Key='Communication_Dataset.zip')['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about
the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
# Unzip the Dataset Zip File

from io import BytesIO

import zipfile

unzip = zipfile.ZipFile(BytesIO(streaming_body_1.read()), 'r')

file_paths = unzip.namelist()

for path in file_paths:

    unzip.extract(path)

%%bash
```

```
ls Communication_Dataset

# Training Dataset
x_train=train_datagen.flow_from_directory(r'/home/wsuser/work/Communication_Da
taset/training_set',target_size=(64,64),
class_mode='categorical',batch_size=900)

# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/home/wsuser/work/Communication_Data
set/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)

print("Len x-train : ", len(x_train))

print("Len x-test : ", len(x_test))

# The Class Indices in Training Dataset
x_train.class_indices
```

Model creation

```
# Importing Libraries
from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense

# Creating Model
model=Sequential()

# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))

model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Flatten())

# Adding Hidden Layers
model.add(Dense(300,activation='relu'))

model.add(Dense(150,activation='relu'))

# Adding Output Layer
model.add(Dense(9,activation='softmax'))

# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accur
acy'])
```

```
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=len(x_train),epochs=10,validation_
data=x_test,validation_steps=len(x_test))
```

Saving the model

```
model.save('IBM_Communication_Model.h5')
# Current accuracy is 0.8154
# Convert the Saved Model to a Tar Compressed Format
!tar -zcvf IBM_TrainedModel.tgz IBM_Communication_Model.h5

%%bash


from ibm_watson_machine_learning import APIClient
wml_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "mNVF7E95G-awR213njShj1GiUfN-1SpPq-ko8Wx7na1-"
}

client = APIClient(wml_credentials)


def guid_from_space_name(client, space_name):

    space = client.spaces.get_details()

    return (next(item for item in space['resources'] if item['entity']["name"]
== space_name)['metadata']['id'])

space_uid = guid_from_space_name(client, 'communication_model_deployment')

print("Space UID : ", space_uid)

client.set.default_space(space_uid)

client.software_specifications.list()


software_spec_uid =
client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")

software_spec_uid

model_details = client.repository.store_model(model='IBM_TrainedModel.tgz',
meta_props={
```

```
    client.repository.ModelMetaNames.NAME: "CNN",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_spec_uid,
    client.repository.ModelMetaNames.TYPE: "tensorflow_2.7"})
model_id = client.repository.get_model_uid(model_details)
```

GitHub & Project Demo Link

Github link

https://github.com/IBM-EPBL/IBM-Project-4232-1658725436

Project Demo Link :

https://drive.google.com/file/d/1pZXe5sPETg04x28Hjlq1-jcm7Mmjc5KS/view?usp=drivesdk